01

Joystick

ROS

WeGo

Joystick_spec

- 무선연결 : 2.4 Ghz
- 피드백 : 듀얼 진동
- D-패드 : 4-스위치
- 배터리 : AA 2개

Joystick_control

	전진	후진	좌회전	우회전
동작 버튼	L1+왼쪽스틱 (위 방향)	L1+왼쪽스틱 (아래 방향)	L1+오른쪽스틱 (위 방향)	L1+오른쪽스틱 (아래 방향)

Joystick_포트확인

- 조이스틱 포트 확인

  $ ls /dev/input

조이스틱을 이용한 이동값 확인

  $ sudo jstest /dev/input/jsx (x에 확인된 포트 번호 입력)

Joystick_사용방법

- 조이스틱 포트 확인

  $ ls /dev/input

조이스틱을 이용한 이동값 확인

  $ sudo jstest /dev/input/jsx (x에 확인된 포트 번호 입력)

Joystick_package install

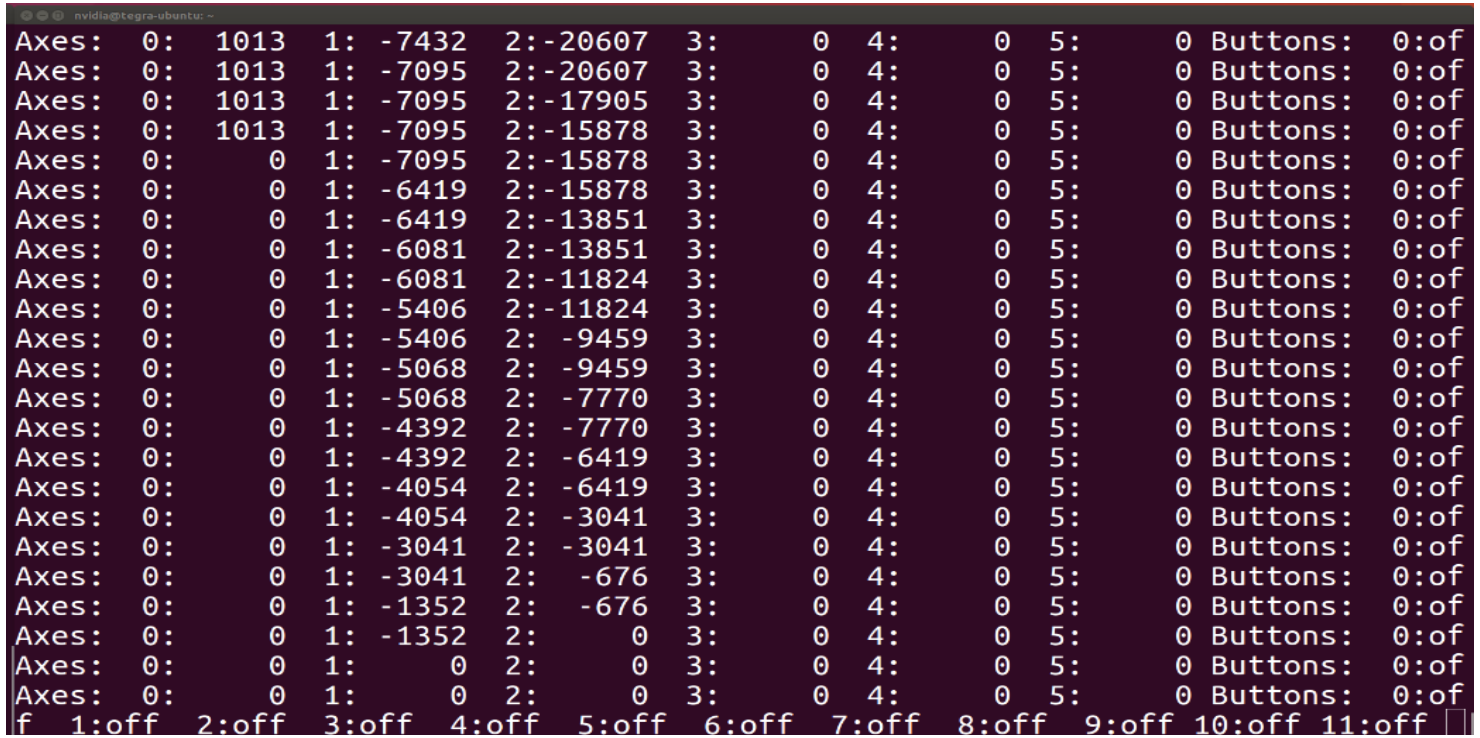
- $ sudo apt-get install joystick

~/racecar-ws/src/racecar/racecar/config/racecar-v2/joy_teleop.yaml

Joystick 노드 정보, 파라미터 값 확인 가능
(racecar 패키지 설치한 경우)

Joystick_사용방법

- 조이스틱 입력 장치의 변경_data (올바른 장치 입력 경우)



02

Joystick Control

F710 조이스틱을 이용한 제어

- 조이스틱 “ SWITCH ‘D’ ”

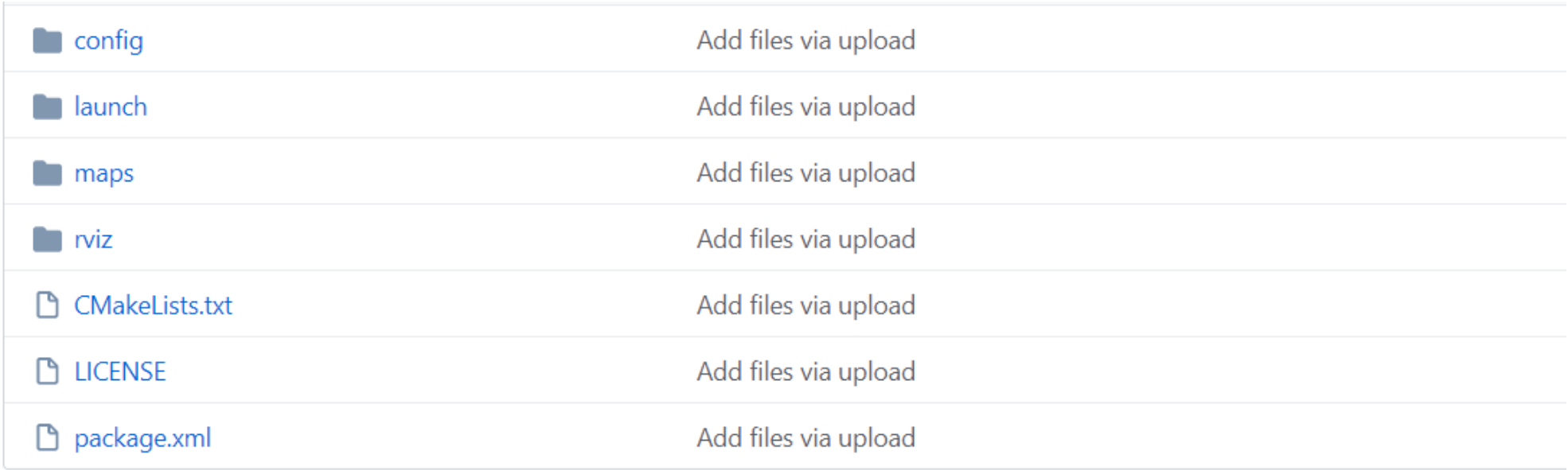
```
$ cd catkin_ws

$ source devel/setup.bash

$ roslaunch racecar teleop.launch
```

ROS

WeGo

Racecar_Software install

- git clone https://github.com/joosun94/TX2_Racecar-Software.git



ROS

WeGo

F710 조이스틱을 이용한 제어

- 조이스틱 axis (변경)

```yaml
joy_node:
  deadzone: 0.01
  autorepeat_rate: 20
  coalesce_interval: 0.01

teleop:
  ackermann:
    type: topic
    message_type: ackermann_msgs/AckermannDriveStamped
    topic_name: ackermann_cmd_mux/input/teleop
    deadman_buttons: [4]
    axis_mappings:
      -
        axis: 1
        target: drive.speed
        scale: 2.0                  # joystick will command plus or minus 2 meters / second
        offset: 0.0
      -
        axis: 3
        target: drive.steering_angle
        scale: 0.34                 # joystick will command plus or minus ~20 degrees steering angle
        offset: 0.0
```

ROS

WeGo

VESC & RPLiDAR_Application

ROS Package_Problem Sensor응용

RPLiDAR를 이용하여 정지하는 Python Code 작성

1M : 이하일 경우, 터미널 창에 데이터를 “STOP”이라고 표시
[모터 정지]

1.5M : 이상일 경우, ‘Auto Move’ 표시
[모터 구동]